REMARKS

Claims 1-16 are pending in the present application. Reconsideration of the claims is respectfully requested.

I. 35 U.S.C. § 103, Obviousness

The examiner has rejected claims 1-16 under 35 U.S.C. § 103(a) as being unpatentable over *Trueblood* (U.S. 5,748,499) in view of *Wong et al.* (US 6,216,152). This rejection is respectfully traversed.

Claim 1 of the present application reads:

> 1. A method for a distributed audio server, the method comprising the computer-implemented steps of:
>
> generating audio data and graphic data in a platform-independent application;
>
> sending the graphic data to a display server on a client machine specified by a display environment variable; and
>
> sending the audio data to a platform-independent audio server on the client machine specified by an audio environment variable or by an audio command line parameter.

Claim 14 recites similar limitations for a computer program product.

In rejecting the claims, the Examiner states:

> As per claims 1 and 14, Trueblood teaches a method for a distributed audio server (column 2, lines 43-49), the method comprising the computer implemented steps of: generating audio data and graphic data (abstract); sending the graphic data to a display server on a client machine specified by a display environment variable (abstract); and sending the audio data to an audio server on the client machine specified by an audio environment variable or by an audio command line parameter (abstract). (Office Action, dated 4/25/02, page 2)

A prima facie case of obviousness is established when the teachings of the prior art itself suggest the claimed subject matter to a person of ordinary skill in the art. *In re Bell*, 991 F.2d 781, 783, 26 U.S.P.Q.2d 1529, 1531 (Fed. Cir. 1993). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). In comparing *Trueblood* to

the claimed invention to determine obviousness, limitations of the presently claimed invention may not be ignored.

Trueblood teaches a method and system for recording and playing back X-Window command streams. It has no bearing on a method for sending platform-independent audio data to a client machine with a platform-independent audio server defined by an audio environment variable.

Trueblood teaches a method and system for recording command streams from a user at set intervals and the ability to replay those command streams by means of a GUI that simulates a VCR. More specifically, *Trueblood* teaches a system for recording air traffic control commands, and the ability to replay and review such data. None of these features relate to the limitations of claims 1 and 14. *Trueblood* does not disclose sending platform-independent audio data to an audio server on a client machine. The only mention of audio data relates to a time-stamped audio track that can be replayed. This is not the same as a platform-independent audio server and does not cover the limitations of the present invention. Nor does *Trueblood* teach an audio environment variable or command line parameter for specifying the audio server operating on the client. In fact, *Trueblood* teaches:

> In the present invention, the graphics software is platform-independent and can operate on any work station embodying an appropriate operating system. However, when audio is included, the audio client communicates with the audio hardware of the work station, thus making the overall system no longer platform-independent. (Col. 8, lines 6-11)

Furthermore, the X-Windows protocol used in the *Trueblood* system cannot support platform-independent audio data streams, which is why *Trueblood* must rely on the platform-dependent audio playback method described above. This limitation of the X-Windows system is specifically pointed out in the present application:

> When a Java application is executing on a Unix host machine through X Windows, the graphics generated by the Java application may be distributed using the X Windows protocol to an X Windows server that is running on a client machine.
>
> The Java runtime environment contains an audio playback engine that uses the native audio support of the supporting machine to play the audio data. However, there is no support for audio in the X Windows protocol. In most cases, when a Java application on a host machine

> generates an audio datastream, the underlying platform-specific operating system will employ the audio support of the underlying machine so that the audio will be audible at the underlying host machine and not the client machine. Hence, if a user at a client machine remotely executes a Java application on a host machine, the graphics for the Java application will appear on the user's client machine, but the audio will be played on the remote host machine, and the user will not hear the generated audio on the client machine. This situation is possibly unexpected to the user and certainly undesirable. (Specification, page 2, line 27-page 3, line15)

No part of *Trueblood* teaches or suggests a method to overcome this limitation of the X-Windows protocol. By contrast, the limitations of claims 1 and 14 do overcome this limitation in the X-Windows protocol.

With regard to platform independence, the Examiner states:

> Trueblood fails to teach that the application is platform independent. Wong et al. teach a media plug-in application that is platform independent (abstract; column 5, lines 30-51). It would be obvious to one of ordinary still in the art at the time the invention was made to employ Wong's teachings within the system of Trueblood because using a platform independent application would allow it to run on various types of machines. (Office Action, page 2)

The simple fact that the plug-in decoders taught in *Wong* are platform-independent plug-ins does not mean that they can be combined with *Trueblood* to produce the limitations of the present invention. In determining obviousness, an applicant's teachings may not be read into the prior art. *Panduit Corp. v. Denison Mfg. Co.*, 810 F.2d 1561, 1575 n. 29, 1 U.S.P.Q. 1593, 1602 n. 29 (Fed. Cir. 1987) (citing need to "guard against hindsight and the temptation to read the inventor's teachings into the prior art").

The plug-ins taught by *Wong* are designed to be used with applications, most notably web browsers. However, the present invention does not rely on applications such as web browsers. Instead, the graphics and audio data in the present invention are sent to a display server and audio server. Servers are fundamentally different than the web browsers that use the plug-ins taught by *Wong*. Browsers actively search for data, whereas servers received data passively as it is sent to them. *Wong* does not teach or suggest the use of its plug-in decoders with servers, such as the display and audio servers in the present

invention. In addition, it is unclear if it is even technically feasible to apply the plug-ins in *Wong* with servers, such as those taught by the present invention.

Furthermore, the mere addition of plug-ins is not sufficient to overcome the inability of the *Trueblood* system to support platform-independent audio data streams, which can be done with the present invention, and it certainly does not produce a system that can support platform-independent audio data by means of a platform-independent audio server on a client machine. Therefore, the proposed combination of *Trueblood* and *Wong* does not produce the limitations of claims 1 and 14.

Therefore, the rejection of claims 1 and 14 under 35 U.S.C. § 103(a) has been overcome.

In rejecting the other independent claims, the Examiner states:

> As per claims 4 and 9, Trueblood teaches a method for a distributed audio server (column 2, lines 43-49), the method comprising the computer implemented steps of generating audio data (abstract). Trueblood fails to teach that the audio server is implemented as a platform-independent application. Wong et al. teach a media plug-in application that is platform independent (abstract; column 5, lines 30-51). It would be obvious to one of ordinary skill in the art at the time the invention was made to employ Wong's teachings within the system of Trueblood because using a platform independent application would allow it to run on various types of machines.
>
> The method of determining whether an audio environment variable or an audio command line parameter is defined and if an audio environment variable or an audio command line is defined, sending the audio data to a platform-independent audio server on a client machine specified by the audio environment variable or by the audio command line parameter is inherent to Trueblood's invention (column 4, lines 55-67), the method is more further disclosed by Wong et al. (column 7, lines 1-37). Because Wong et al. refer to the method being applied to media, the method inherently includes audio data and graphic data. This provides the basis on which claims 5 and 10 are rejected. (Office Action, page 3)

This rejection is traversed for the same reasons explained in regard to claims 1 and 14. Furthermore, the Applicant disputes with the Examiner's assertion that the use on audio server and audio environment variable are inherent to *Trueblood* and *Wong*. As explained above, *Trueblood* teaches a system that operates using X-Windows, which does not support audio. Client machines using the X-Windows protocol cannot support platform-independent audio data streams without the local audio environment variable (or

command line parameter) and audio server provided by the present invention. It is well known in the art that the audio server and audio environment variable are not inherent parts of the X-Windows protocol. Again, no part of *Trueblood* teaches or suggests a method to overcome this limitation of X-Windows. Therefore, the only thing inherent about X-Windows system in *Trueblood* is its *inability* to support platform-independent audio data.

As for *Wong*, the use of audio and video media does not inherently include the use of an audio environment variable and audio server or a display server, which are distinct from plug-in decoders. Moreover, as stated above, the plug-ins in *Wong* are used with applications such as web browsers, which are fundamentally distinct from the display and audio servers in the present invention.

Therefore, the rejection of claims 4, 5, 9, and 10 under 35 U.S.C. § 103(a) has been overcome.

Since claims 2-3, 6-8, 11-13, and 15-16 depend from the independent claims and contain the limitations of the independent claims, they are distinguished from *Trueblood* and *Wong* for the reasons explained in regard to claims 1, 4, 9, and 14.

Therefore, the rejection of claims 1-16 under 35 U.S.C. § 103(a) has been overcome.

II. Conclusion

It is respectfully urged that the subject application is patentable over *Trueblood* and *Wong* and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: July 19, 2002

Respectfully submitted,

Christopher P. O'Hagan

Christopher P. O'Hagan
Reg. No. 46,966
Carstens, Yee & Cahoon, LLP
P.O. Box 802334
Dallas, TX 75380
(972) 367-2001
Attorney for Applicant